Data Analytics for Info System IS665

Shivani Bhikadia
Sri Shantan Palwayi
Tanvi Malali

Fraud Detection using ML

First Steps

We input the data (train_transaction.csv and train_identity.csv) and import all the libraries we need; i.e., pandas, numpy, matplotlib, seaborn, gc, and sklearn. We check the heads of the two csv input files, and the shapes. The two training sets were then merged on the basis of the column TransactionID, so we can work on one joint training set. A function is used to reduce memory usage. We then plot the frequency of the IsFraud binary column, and notice that a very small amount of transactions are fraudulent.

EDA (Exploratory Data Analysis)

We now begin to analyse each column and check its connection with the IsFraud column.

- TransactionDT - The description of rows that have IsFraud with value 1 and IsFraud with value 0 are obtained; and the frequency of TransactionDT is plotted using seaborn.
- TransactionAmt - The non - fraudulent rows with IsFraud equal to 0 is obtained, and from this, the description of TransactionAmt is obtained. The fraudulent rows are treated in the same way. The average transaction amounts are printed and checked for outliers. Two outliers are found, and are duplicate, and are dropped.
- The transactions which have transaction amount greater than 10000 are dropped, and created a new feature called LogTransactionAmt to reduce the imbalance from the original column, and TransactionAmt is dropped.

- The log transaction amount column is plotted for both fraudulent and legit transactions. From this, the log transaction amount range of 3-5 has the most chance of legit transaction, and overlap values are higher after 5 and before 3.
- Transaction Amount and Transaction DT are plotted using seaborn and checked for IsFraud values.
- ProductCD - To check which products have the most fraud, we group the columns by IsFraud and ProductCD and plot the data, and observe that there is a high chance of fraud for product C, followed by W,H,R and S.
- After analysing the different card types, discover is shown to have the most fraud, followed by visa and mastercard; and it is also discovered that credit card users tend to face more fraud. The columns Charge card and credit or debit card have no effect, and are replaced as one column.
- Analysing the address columns, most of the data is found to be from a country with code 87.
- Analysing the email columns, gmail and hotmail are found to have the most data, and consequently the most fraudulent transactions.
- Correlations are found in the C and D columns, and will be checked later in modelling.
- NaN values need to be dealt with in the V columns; and M columns have some correlations.
- DeviceInfo and DeviceType are analysed and it is found that mobile transactions are mostly fraudulent, after plotting DeviceType.

Feature Engineering and Selection

Categorical values are determined, and two columns with mostly NaN values, id_23 and id_27 are dropped. Other NaN values found are replaced with the value -999, which is least disruptive. Categorical values found in train_identity are replaced with numerical values. We use Typecasting to convert all categorical values to type string, so there isn't any discrepancy, and use Label encoding to convert categorical values to numerical values that can be used in prediction by the machine learning model.

We then use downsampling to reduce values from the dataset to balance it, as if we use the whole dataset, our model will be overfitted; and will be biased. Both data frames are merged on the TransactionID column, since they have it in common, and the final train dataset is ready for modelling.

Modelling

We now choose a model and fit the model using our train dataset, and predict the IsFraud values, and test the accuracy of the model using the AUC (Area Under the roc Curve). The ROC (Receiver Operating Characteristics) curve is a graph showing the performance of a model at all thresholds.

- From sklearn, import train test split, and assign X as all the rows under all columns except IsFraud, and y as all the rows under the column IsFraud.
- X_train, X_test, y_train and y_test are assigned using the train test split function, with X_train containing 80% of the rows, X_test containing 20% of the rows, y_train containing 80% of the IsFraud column, and y_test containing 20% of the IsFraud column.
- We first choose the Decision tree Classifier model, to test the accuracy of our data. X_train and y_train are fit, and the predicted results are assigned to y_pred. The accuracy of the test is around 84%. It's a good result, but we can do better using another model.
- Random forest Classifier model is chosen as it is perfect for binary targets, i.e., the IsFraud column; it is perfect for big datasets; and it provides reasonable predictions. It also corrects overfitting.
- X_train and y_train are fit, and the same model is run with the X_test data, and the results are assigned to y_pred.
- The AUC score is found to be around 95%, which is our final submission.